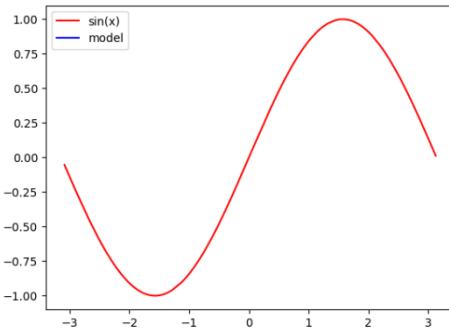
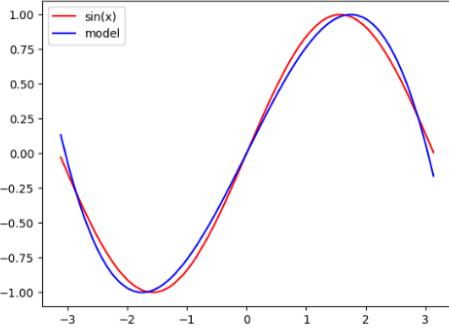
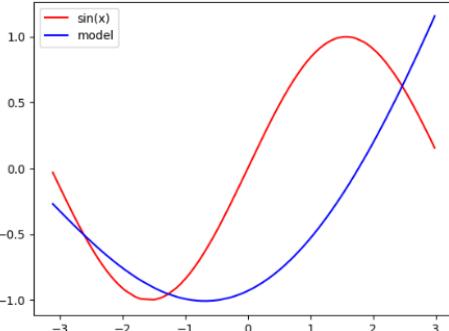
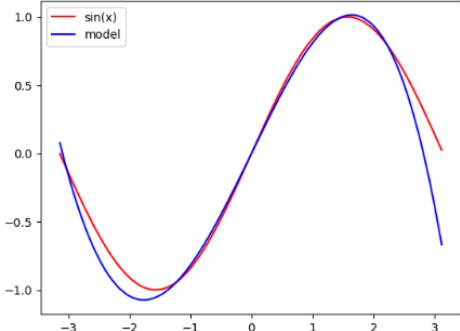
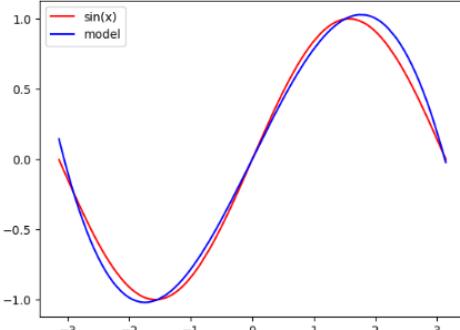
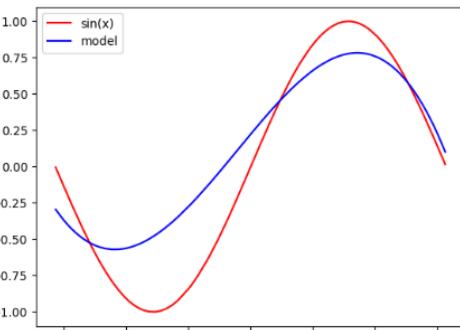


作业二：神经网络初尝试

实验结果

损失函数	学习率	模型多项式	图形
MSELoss	1e-4	y = nan + nan x + nan x^2 + nan x^3 + nan x^4 ? + nan x^5 ?	
	1e-5	y = 0.001037392416037 6191 + 0.857995092868804 9 x + - 0.000750421080738 306 x^2 + - 0.093811921775341 03 x^3 + 0.000109740234620 4035 x^4 ? + 0.000109740234620 4035 x^5 ?	
	1e-6	y = - 0.930618584156036 4 + 0.227599874138832 1 x + 0.171805575489997 86 x^2 + 0.002423509955406 189 x^3 + - 0.002357216551899 91 x^4 ? + - 0.002357216551899 91 x^5 ?	

L1Loss	1e-4	$y = -$ 0.000565071590244 77 + 0.920507192611694 3 x + - 0.000524542003404 3491 x^2 + - 0.106410749256610 87 x^3 + - 0.003288253210484 9815 x^4 ? + - 0.003288253210484 9815 x^5 ?	
	1e-5	$y =$ 0.001288067316636 4431 + 0.883130311965942 4 x + - 0.001038566464558 2438 x^2 + - 0.099545300006866 46 x^3 + 0.000739783339668 0653 x^4 ? + 0.000739783339668 0653 x^5 ?	
	1e-6	$y =$ 0.223834574222564 7 + 0.514645457267761 2 x + - 0.029949137941002 846 x^2 + - 0.046071749180555 344 x^3 + - 0.000324835331412 03225 x^4 ? + - 0.000324835331412 03225 x^5 ?	

实验结果分析

损失函数与学习率的影响：

本次实验使用了两种损失函数（`MSELoss` 和 `L1Loss`），并且尝试了不同的学习率（`1e-4`、`1e-5`、`1e-6`）。不同的损失函数和学习率组合对模型参数的优化过程产生了显著影响。

- **`MSELoss`:**

- 学习率为 **`1e-4`** 时，模型的参数结果出现了 `NaN`，是由于梯度爆炸或学习率过大引起的。
- 学习率为 **`1e-5`** 时，模型成功地收敛，参数值接近了理想的多项式形式。参数的调整体现了模型对训练数据的逐步逼近。
- 学习率为 **`1e-6`** 时，虽然结果依然接近正确的多项式形式，但学习率过低，收敛速度较慢，影响了模型的表现。

- **`L1Loss`:**

- 学习率为 **`1e-4`** 时，损失函数相对较大，且参数值表现出不稳定性，可能是由于梯度的不均衡导致。
- 学习率为 **`1e-5`** 时，`L1Loss` 的表现较为稳定，参数趋于收敛，且模型的拟合能力增强。
- 学习率为 **`1e-6`** 时，但学习率过低，参数的收敛速度极慢，模型未充分优化。

总结：

这些实验结果显示出了损失函数和学习率对神经网络训练过程的关键影响。`MSELoss` 一般能够更好地拟合平滑的数据，但当学习率过大时，会导致参数更新不稳定；而 `L1Loss` 在面对某些数据时能更好地减少异常值的影响，但也可能导致收敛速度较慢。